



HAND GESTURE RECOGNITION FOR CONTROLLING MOUSE

Narayanpure Rohit Babureddy¹, Preetham D², Vinay B T³, Sudeep J⁴

¹⁻³Student, NIE Institute of Technology, Mysore, Karnataka

⁴Assistant Professor, NIE Institute of Technology, Mysore, Karnataka

Abstract: Hand gesture recognition has been a very interesting aspect in the field of Human Computer Interface due to its flexibility and user friendly. The gesture recognition technique is being used to develop different types of system to develop communication among disabled people or a system for controlling a device. Major challenges for developing a efficient hand gesture recognition techniques are non – uniform backgrounds, difference in the size and shape of user’s hand, different hand gesture types such as static and dynamic gestures.

Mouse plays a very important role as an input device in the Human Computer Interaction (HCI). There have been different techniques introduced to replace the functionalities of mouse. Different methods such as using hand gloves, coloured finger tags, accelerometer, Gyroscope, Bluetooth, etc. In the system we are using different types of dynamic gestures to replace the mouse functionalities without using any type of external devices and using only the system camera with the help of mediapipe and opencv.

Keywords: Hand Gestures, keypoints, Mediapipe, OpenCV

I. INTRODUCTION

Recently, computer-assisted communication technology (HCI) has attracted attention as a promising means of human computer communication. The interest in human gesture communication with HCI has also increased. This includes the movement of human hands. Human hand gestures are verbal communication from simple pointing to complex interactions between people. The hand gesture-based virtual mouse is a software program that allows users to provide mouse inputs to a device without using a physical mouse. This research presents a computer creative hand gesture based virtual mouse device that produces, using hand gestures and hand key points detection, for performing mouse activities on the computer. The major purpose of the suggested device is to perform laptop mouse cursor functions using a webcam or a built-in digital camera within the laptop rather than a conventional mouse device. In research papers, they used external devices like hand gloves and web camera, they used colour detection algorithms to detect gestures and to perform functions.

In our paper we have used python programming, opencv and media pipe package to give high performance. The hand gesture recognition using the open Python programming language, as well as OpenCV, a computer vision package that is employed inside the system. As a result, the Media Pipe package is used to track the hands key points and monitor the movements of hand in this project.

The web camera gathers and approaches the collected frames in this system, identifying various hand motions and hand tip gestures, and then performing real mouse functionalities. Our aim is to give virtual mouse using hand gestures with high accuracy, easy to use and to perform in long range.

II. LITERATURE SURVEY

Gesture recognition is an important topic in computer vision because of its wide range of applications, such as Human computer Interaction, sign language interpretation, and visual surveillance. 3-D gesture recognition has attracted a lot of research interests in computer vision, pattern recognition, and human-computer interaction. The rising use of depth sensors greatly increases various hand gesture recognition approaches and applications, which were severely limited in the 2D domain with conventional cameras. Conventional approach gesture control in automobile industry etc. possess certain challenges such as high computational power ,environmental lighting constraints or integrating voice control commands in noisy environments has also certain challenges, which can be solved and can be overcome by the use of Gesture Recognition Control using Time of flight camera which is fast and efficient. This paper aims at the use PicoZense ToF camera sensor and OpenCV libraries; we can easily overcome the challenges of recognising the gestures in low lighting conditions with noisy environment.^[1]

Computer vision is the field of Artificial Intelligence that plays an important role in Hand Gesture Recognition by understanding the digital images or videos to do a predefined operation. Here, the input data is taken from the live



feed of a webcam to detect the image. The live video is captured by taking the set of image frames. Then we subtract the foreground from the background of all the images to get the binary image with the maximum noise removal. Next, we form the contour to detect the hand region from the original frames. We apply Canny edge algorithm to calculate the edges of the input data to detect the contours. Using this we form the convex hull on hand contour to obtain the Region of Interest. We use the hybrid model of Jarvis March and Graham Scan Algorithms to achieve the convex hull formation. Based on the convex hull, we calculate the defect points, find the centre of the region of interest (ROI), generating the circle on ROI, bitwise AND operation for the circle and ROI, and determine the number of fingers shown. The extracted information until this step stands as a key feature to upgrade for dynamic gestures. With the help of Machine Learning techniques, we recognize complex motion gestures to perform operations on computers. depicts our work flow.^[2]

In this paper, they design an algorithm to detect the fingers, identify the gestures and to control the operations of a mouse. This paper consists of three sub-sections namely color identification, gesture identification and virtual mouse control along with the overall system description. In this paper, they use the Python language for the implementation of algorithm. We use the OpenCV library for image processing and pyautogui library for mouse control. The algorithm use two types of methods for implementation of mouse control. One includes the usage of color caps and other uses the recognition of bare hand gesture recognition. A method for on-screen cursor control without any physical connection to a sensor is presented. Identification of colored caps on the fingertips and their tracking is involved in this work. Different hand gestures can be replaced in place of colored caps for the same purpose. Different operations of mouse controlled are single left click, double left click, right click and scrolling. Various combinations of the colored caps are used for different operations. Range of skin color can be varied in the program in accordance with the person to be used, surrounding lightening conditions. An approximate area ratio that is not being used by the hand in the convex hull is taken after analyzing the program output at different gestures of the hand.^[3]

III. METHODOLOGY

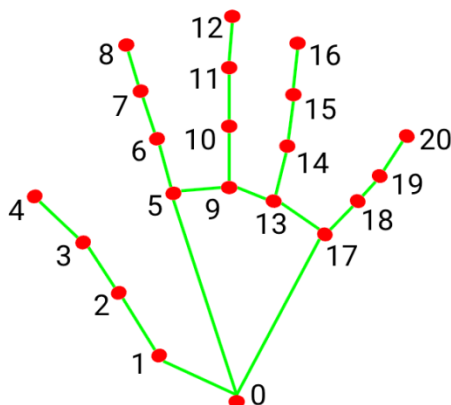
Proposed System:

The proposed system aims to use Python language and media pipe along with OpenCV technology for controlling mouse functions using hands. The model mainly includes 8 functions,

1. Brightness Control
2. Volume Control
3. Left Click
4. Right Click
5. Cursor control
6. Drag and Drop
7. Multiple item selection
8. Double Click

Methodology:

1. first we will define hand Gesture Encodings as per the mediapipe hand key points and also we will define Multi-handedness Labels.
2. Developing the function named Convert Mediapipe Landmarks to recognizable Gestures. This is phase where we will identify or recognize the hand gesture.
3. Based on the hands landmarks we will construct or define different functionality



- | | |
|-----------------------|-----------------------|
| 0. WRIST | 11. MIDDLE_FINGER_DIP |
| 1. THUMB_CMC | 12. MIDDLE_FINGER_TIP |
| 2. THUMB_MCP | 13. RING_FINGER_MCP |
| 3. THUMB_IP | 14. RING_FINGER_PIP |
| 4. THUMB_TIP | 15. RING_FINGER_DIP |
| 5. INDEX_FINGER_MCP | 16. RING_FINGER_TIP |
| 6. INDEX_FINGER_PIP | 17. PINKY_MCP |
| 7. INDEX_FINGER_DIP | 18. PINKY_PIP |
| 8. INDEX_FINGER_TIP | 19. PINKY_DIP |
| 9. MIDDLE_FINGER_MCP | 20. PINKY_TIP |
| 10. MIDDLE_FINGER_PIP | |



we have used pyautogui packages for executing mouse functions.

we will read the frame from real time web cam please find the info below:

We create a VideoCapture object and pass an argument '0'. It is the camera ID of the system. In this case, we have 1 webcam connected with the system. If you have multiple webcams then change the argument according to your camera ID. Otherwise, leave it default

GestureController.cap = cv2.VideoCapture(0) function will start the camera.

The cap.read() function reads each frame from the webcam.

mpDraw.draw_landmarks() function we draw all the hand landmarks in the frame.

The get_position functionality where we will look into Region of interest(ROI), cropROI, find_gesture, Tracker, GestureController.

pyautogui.position() is used to execute the get_position function

set_finger_state function is used to find gesture Encoding using current finger_state.

Finger_state = 1 if finger is open, else 0 if finger is closed.

The get_gesture function is used to handle fluctuation due to noise.

We have written functions for each hand gesture recognition.

IV. IMPLEMENTATION

To implement a system successfully, a large number of inter-related tasks need to be carried out in an appropriate sequence. Utilizing a well-proven implementation methodology and enlisting professional advice can help but often it is the number of tasks, poor planning and inadequate resourcing that causes problems with an implementation project, rather than any the tasks being particularly difficult. Similarly with cultural issues it is often the lack of adequate consultation and two-way communication that inhibits achievement of the desired results. In this project we concentrate on executing the mouse commands without being in any contact with the mouse.

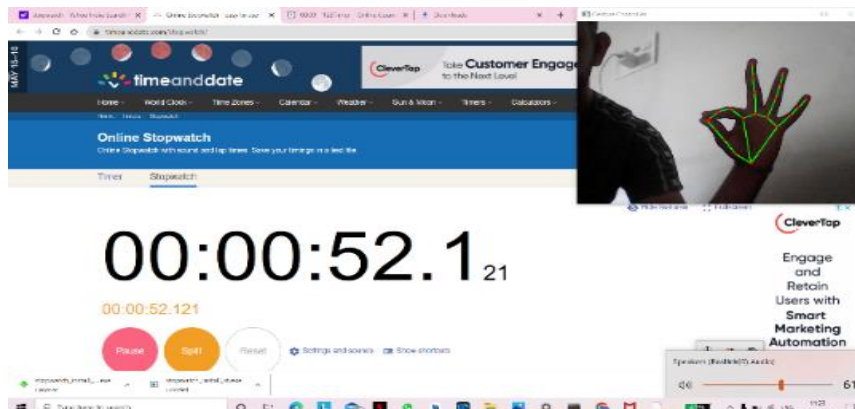
This can be achieved with the help of the commonly used hand gestures and any type of web-camera or preinstalled camera. With the help of the python language and its libraries this can be achieved easily. With the help of mediapipe and opencv technology the video would be captured and converted as the systems need. Python package pyautogui is used to simulate the mouse commands and to write the essential code for each command. By using the python package pycaw.pycaw the systems volume command would be easily controlled. Once after starting the application it will at first start recognizing the hand and also would be calculating for the assigned keypoints for the hand.

For each command there are different types of gestures specified. The calculation is done by calculating the distance between two points. To control the brightness and volume is done with the help of pinch which is by bringing the thumb and finger close i.e., 4 and 8 point closer. Similarly by using the V gest we can do the cursor control which will move the cursor as we will move our hand and also double click by closing the V gest. Left click and right click by closing and opening the fingers.

V. RESULT

1. Volume and Brightness Control

As mentioned earlier in the methodology for volume control we would be using the pinch gesture for increasing and decreasing the volume and brightness. If we move the pinch horizontally the brightness would be controlled and if vertically the volume would be controlled.





2. Cursor Control :

If we do the V gesture that is by using the index finger and middle finger we can control the cursor movement.



3. Left Click :

In the V gesture if we close down the index finger and keeping the middle finger we can perform the left click function.



4. Right Click :

In the V gesture if we close down the middle finger and keeping the index finger we can perform the right click.



5. Drag and Drop and Multiple Item selection :

For the drag, drop and multiple Item selection, we can open our palm and close it and accordingly move it to select or drag the item.



VI. CONCLUSION

With the help of the camera the gestures done by the user would be identified and the same would be executed in the computer. For each function different types of gestures are being allocated for a clear execution of commands. Thus the traditionally usage of a wired mouse would be terminated and the cursor movements can be handled very easily. This application is user friendly and easy to upgrade for the future purpose.

VII. REFERENCE

- [1]. International Journal of Applied Engineering Research ISSN 0973-4562 Volume 16, Number 4 (2021) pp. 302-307 © Research India Publications. <https://dx.doi.org/10.37622/IJAER/16.4.2021.302-307> 302 “Hand Gesture Recognition using TOF camera”.
- [2]. Bhavitha B, DivyaPrakash R, Vedha T Selvan, Ramanathan R.”Improved real time approach to static hand gesture recognition”. 2021 11th International Conference on Cloud Computing, Data Science & Engineering.
- [3]. Vantukala VishnuTeja Reddy , Thumma Dhyanchand , Galla Vamsi Krishna , Satish Maheshwaram. “Virtual Mouse Control Using Colored Finger Tips and Hand Gesture Recognition”. 2020 IEEE-HYDCON.
- [4]. Hand Movement Recognition by Using a Touchless Sensor for Controlling Images in Operating Room. Phataratah Sa-nguannarm , Wongwit Senavongse , Theekapun Charoenpong. IEECON 2018, Krabi, Thailand